

VŠB – Technická univerzita Ostrava
Fakulta strojní
Katedra automatizační techniky a řízení

Využití PLC pro řízení tříosého mechanismu
Using PLC for control of three-axis mechanism

Student:

Jakub Novotný

Vedoucí bakalářské práce:

Ing. Jaromír Škuta, PhD

Zadání bakalářské práce

Student: **Jakub Novotný**
Studijní program: B2341 Strojírenství
Studijní obor: 3902R001 Aplikovaná informatika a řízení
Téma: **Využití PLC pro řízení tříosého mechanismu**
Using PLC for Control of Three-axis Mechanism

Zásady pro vypracování:

1. Charakterizujte PLC firmy ABB, prostředím pro programování PLC a s nástroji pro monitorování a konfiguraci.
2. Popište laboratorní modely obráběcích strojů a možnosti jejich připojení k PLC firmy ABB.
3. Realizujte algoritmy polohování pracovního nástroje (pracovního prostoru) vybraného modelu pomocí PLC s následnou možností monitorování. Vytvořte konfigurační a monitorovací aplikaci v PC.
4. Zhodnoťte dosažené výsledky a navrhněte další směr řešení.

Seznam doporučené odborné literatury:

BOYER, S. A. SCADA: *Supervisory Control and Data Acquisition*, 2nd Edition. New York (USA): ISA, 1999. 215 p. ISBN 1-55617-660-0.
CNC výukové kity [online]. HELAGO-CZ, s.r.o. 2002-2012. [cit. 11.10.2012]. Dostupné z: <http://www.helago-cz.cz/catalog/cnc-vyukove-kity/>
VAEPROSYS - AUTOMATIZACE - PLC AC500 - Dokumentace ke stažení [online]. VAE ProSys s.r.o. 2010. [cit. 11.10.2012]. Dostupné z: <http://vaeprosys.cz/list.asp?p=133&k=32&mk=9&ms=32>
VLACH, J. *Počítačová rozhraní, přenos dat a řídicí systémy*. Praha, BEN-technická literatura, 1997, ISBN 80-85940-17-4.
WHITT, M. D. *Successful Instrumentation and Control Systems Design*. New York (USA): ISA, 2003. 360 p. ISBN 1-55617-844-1.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jaromír Škuta, Ph.D.**

Datum zadání: 14.12.2012
Datum odevzdání: 20.05.2013



prof. Ing. Jiří Tůma, CSc.
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.
děkan fakulty

Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

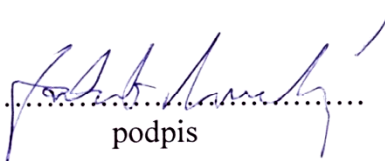
V Ostravě dne 20. 5. 2013

.....
podpis studenta

Prohlašuji, že

- jsem byl seznámen s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- беру на ве́доміі, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě bakalářskou práci užít (§ 35 odst. 3).
- souhlasím s tím, že bakalářská práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího bakalářské práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́доміі, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě: 20. 5. 2013

.....

podpis

Jméno a příjmení autora práce: Jakub Novotný

Adresa trvalého pobytu autora práce: 17. listopadu 32, Havířov – Podlesí, 736 01

ANOTACE BAKALÁŘSKÉ PRÁCE

NOVOTNÝ, J. *Využití PLC pro řízení tříosého mechanismu*. Ostrava, 2013. 45 s.
Bakalářská práce. Strojní fakulta VŠB-TUO. Katedra automatizační techniky a řízení.
Vedoucí ročníkového projektu: Škuta, J.

Tato práce se zabývá využitím PLC při řízení tříosého mechanismu. Jedná se o PLC firmy ABB. Softwarem pro programování tohoto PLC je CODESYS. V práci je popis laboratorních modelů CNC obráběcích strojů dostupných na katedře a popis použité frézky, jejích parametrů a propojení frézky s PLC a PC. Propojení PLC a frézky je realizováno spínacím výkonovým obvodem. Dále tato práce obsahuje příklad využití tohoto systému v podobě řešení algoritmu pro frézování drážky pro těsné pero. Vzájemný pohyb obrobku a frézy je vykonáván pomocí krokových motorů. Součástí práce je i ovládací a monitorovací aplikace.

ANOTATION OF BACHELOR THESIS

NOVOTNÝ, J. *Using PLC for control of three-axis mechanism*. Ostrava, 2013. 45 p.
Bachelor Thesis. Faculty of Mechanical Engineering VŠB-TUO. Department of Control Systems and Instrumentation. Thesis Supervisor: Škuta, J.

This thesis is dealing with utilization of the PLC for control of triaxial mechanism. The PLC is from ABB company. Software for programming this PLC is CODESYS. Here is description of laboratory models of CNC machine tools available on the department and description of milling machine that was used, its parameters and connection of this milling machine with the PLC and PC. Connection of PLC and milling machine is realised with switching amplifying circuit. Then this thesis contains an example of utilization of this system in the form of solution of algorithm for milling a keyseat for sunk key. Mutual movement of workpiece and milling cutter is performed by stepper motors. Application for monitoring and control is also part of this thesis.

Obsah

Seznam použitých značek a symbolů.....	8
Úvod.....	9
1 Programovací prostředí firmy ABB	10
1.1 Programovací jazyky	10
1.2 Monitorování	14
2 CNC výukové kity The Cool Tool	16
2.1 CNC modely dostupné na katedře	16
2.1.1 Unimat celokovový CNC soustruh.....	16
2.1.2 Unimat celokovová 3-osá CNC frézka.....	16
2.1.3 Unimat celokovový 4-osý CNC frézovací systém	17
2.1.4 Unimat celokovová 4-osá turn/mill sestava	18
2.1.5 Portálová frézka Unistep 540/70	18
2.2 Pohonná jednotka - krokový motor	19
2.2.1 Rozdělení dle konstrukčního provedení	19
2.2.2 Princip krokového motoru	19
2.2.3 Funkční části CNC frézky	20
3 Propojení PLC, PC a frézky	21
3.1 PLC řady AC500	21
3.2 Propojení PLC s PC	23
3.3 Propojení frézky s PLC.....	24
3.3.1 Návrh spínacího výkonového obvodu	24
3.3.2 Proudové omezení krokového motorku.....	24
3.3.3 Tvorba plošného spoje.....	26
4 Řešení algoritmu pro vybranou úlohu.....	28
4.1 Systémový návrh.....	28
4.2 Generátor pulsů pro krokové motorky	29
4.3 Pohyb krokového motorku.....	30

4.4	Řízení synchronního pohybu krokových motorků.....	31
4.5	Algoritmus pro vyvrtání drážky pro těsné pero	32
5	Ovládací a monitorovací aplikace	35
6	Závěr	36
	Seznam použité literatury	38
	Přílohy	40

Seznam použitých značek a symbolů

ABB	Firma poskytující technologie pro energetiku a automatizaci (ASEA Brown Boveri)
CFC	Jazyk volně propojovaných bloků (Continuous Function Chart)
COM	Standart sériové komunikace
CNC	Počítačové číslicové řízení (Computer Numerical Control)
DC	Stejnoseměrný proud (Direct Current)
FBD	Jazyk funkčního blokového schématu (Function Block Diagram)
I/O	Vstup/výstup (Input/Output)
IL	Instrukční sada (Instruction List)
IP	Internetový protokol
LD	Jazyk žebříčkového diagramu (Ladder Diagram)
M1, M2	Krokové motorky
PC	Osobní počítač (Personal Computer)
PLC	Programovatelný logický automat (Programmable Logic Controller)
RS-232	Standart sériové komunikace
RS-485	Standart sériové komunikace
SFC	Sekvenční funkční diagram (Sequential Function Chart)
ST	Jazyk strukturovaného textu (Structured Text)
UV	Ultrafialové (Ultraviolet)
rpm	otáčky za minutu (Revolutions Per Minute)
B	Šířka drážky
C	Vzdálenost drážky od kraje hřídele
Dv	Průměr frézy
L	Délka drážky
t	Hloubka drážky
tmax	Maximální hloubka zářezu frézy

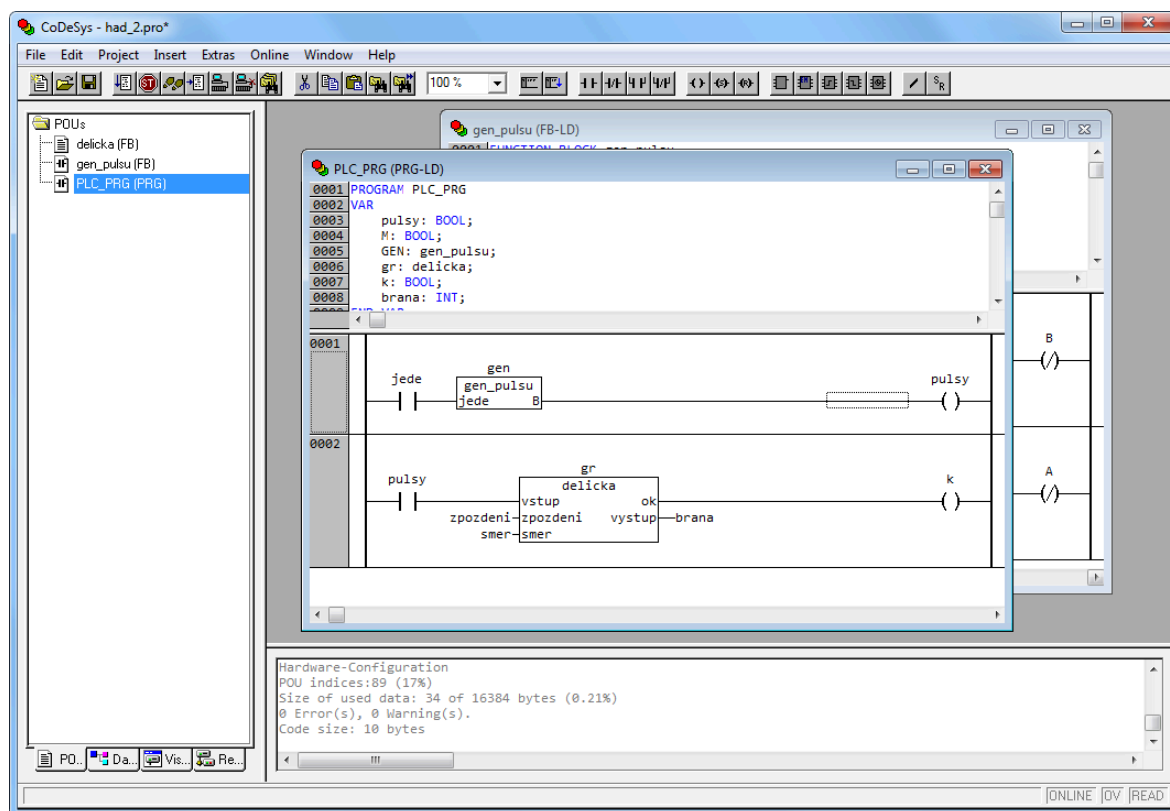
Úvod

Od počátku dějin se lidstvo snaží si v co možná největší míře zjednodušit práci. Na jednu stranu za to může pohodlnost lidí. Na druhou stranu však také díky automatizaci dochází k úspoře lidských zdrojů potřebných pro tvorbu produktu, a tím i k úspoře nákladů. Velká spousta výrobků a strojních částí je vyráběna za pomoci obráběcích strojů. Pro to, aby obráběcí stroj sám fungoval, je pak potřeba mít adekvátní hardwarovou a softwarovou výbavu a samozřejmě také znalosti, jak s nimi zacházet. Díky automatizaci obrábění následně dochází jak k úspoře lidských zdrojů, tak i k úspoře času, za jaký je daný produkt vyroben. Při obrábění je provést vzájemný pohyb obrobku a obráběcího nástroje dle požadované trajektorie. A právě řešením automatického pohybu po určité trajektorii pomocí krokových motorků se tato práce zabývá.

1 Programovací prostředí firmy ABB

Pro programování svých PLC poskytuje firma ABB software firmy 3S-Smart Software Solutions zvaný CODESYS (COntroller DEvelopment SYStem). CODESYS je rozsáhlý softwarový nástroj pro průmyslové automatizační technologie. [Vojáček, A., 2011]

Samotný hlavní program se skládá z několika částí, jak je vidět na Obr. 1.1. Hlavní část prostředí je určená pro okna s programy a pro okna s jednotlivými funkcemi a funkčními bloky. Nahoře je nástrojová lišta, ve které jsou ikony s nástroji pro zrovna aktivní část vytvářeného programu a nástroje pro jeho ovládání. Vlevo jsou karty s nabídkami programových organizačních jednotek, uživatelem vytvořených datových typů, vizualizací a s nabídkami různých nastavení PLC, komunikace, knihoven apod. Dole je pak zápis provedených akcí, případně chyb.

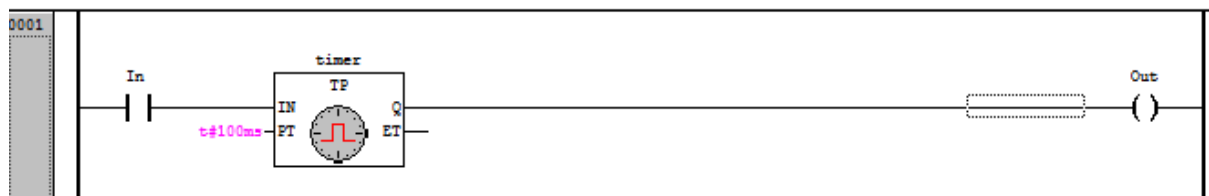


Obr. 1.1- Prostředí programu CODESYS

1.1 Programovací jazyky

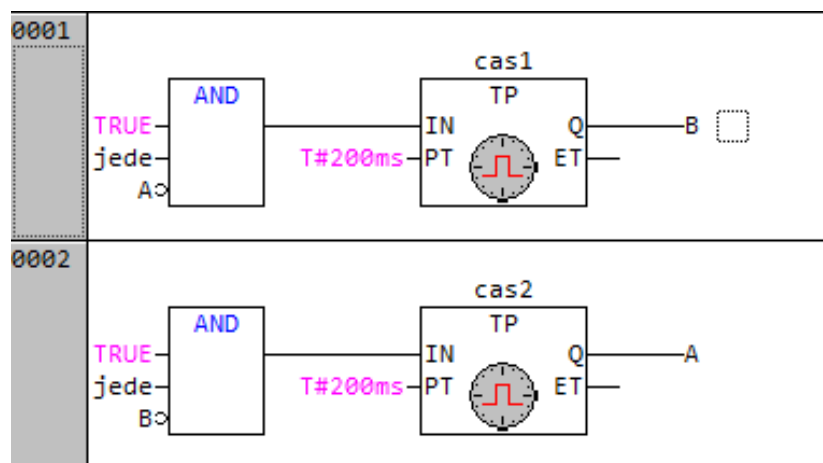
Programy se v prostředí CODESYS dají psát ve více jazycích podle normy IEC 61131-3. Každá jednotlivá část programu může být napsána v jiném jazyce.

Jazyk žebříčkového diagramu LD - Grafický jazyk LD (Ladder Diagram), také nazýván jako jazyk kontaktních schémat, je založen na grafické podobě reléové logiky. Je organizován pomocí sítě propojených grafických prvků. Tato síť je zleva i zprava ohraničená dvěma svislými čarami, nazývanými levá a pravá napájecí sběrnice. Mezi sběrnicemi je tzv. příčka, která může být rozvětvená. Každý úsek této příčky může být ve stavu TRUE nebo FALSE. Do příček mohou být vloženy kontakty, funkce a funkční bloky.



Obr. 1.2 – Příčkový diagram

Jazyk funkčního blokového schématu FBD – Druhý grafický jazyk FBD (Function Block Diagram) pomocí vzájemně propojených grafických bloků vyjadřuje chování funkcí, funkčních bloků a programů. Jde o systém prvků, které zpracovávají signály. Často se zde používají standartní funkční bloky jako např. prvky pro detekci náběžné a sestupné hrany, čítače, časovače apod. Podobně jako u jazyku LD je jazyk FBD uspořádán do jednotlivých řádků. [Martinásková, M., 2004]



Obr. 1.3 - Funkční blokové schéma

Instrukční sada IL – Textový jazyk IL (Instruction List) je tvořen sekvencí instrukcí, z nichž každá začíná na novém řádku. Každá instrukce obsahuje operátor a v závislosti na typu operace jeden nebo více operandů oddělených čárkou. Před instrukcemi může být návěští zakončeno dvojtečkou. Na konci řádku u každé instrukce může být komentář.

0001	LD	TRUE
0002	AND	jede
0003	ST	gener.jede
0004	CAL	gener
0005		
0006	LD	gener.A
0007	ST	pulsy

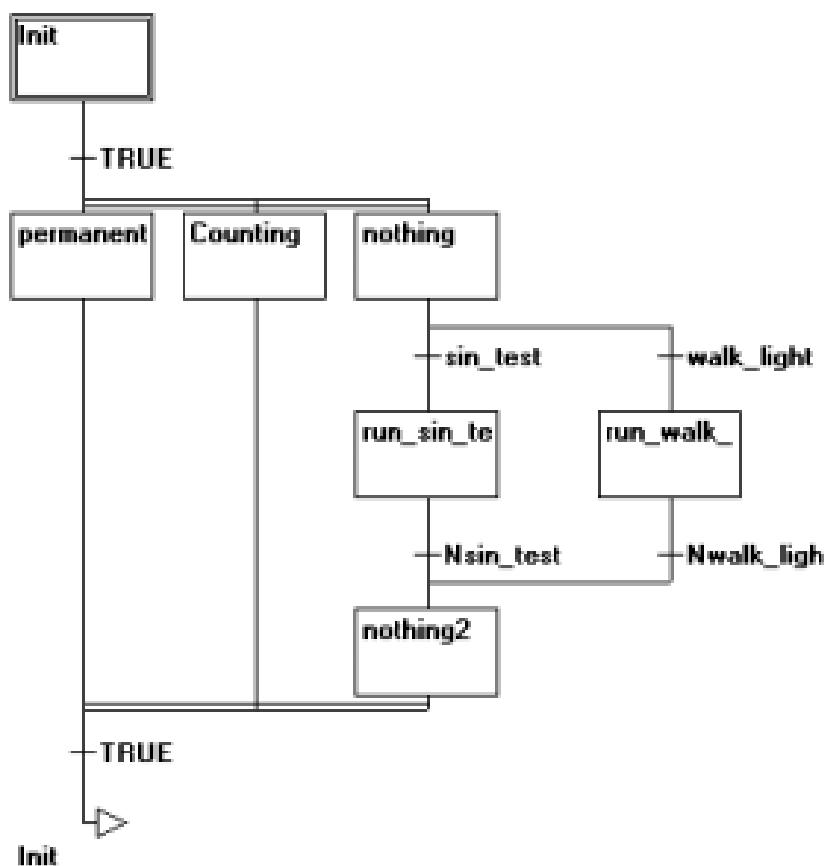
Obr. 1.4 - Seznam instrukcí

Jazyk strukturovaného textu ST – Textový jazyk ST (Structured Text) je výkonný vyšší programovací jazyk, jehož syntaxe je dána povolenými výrazy a příkazy. Vyhodnocením výrazu vyjde hodnota v některém z definovaných datových typů. Na jednom řádku může být více příkazů oddělených středníkem. Tento jazyk je vhodný k tvorbě komplexních funkčních bloků, které pak mohou být použity jiným programovacím jazykem.

0013	IF vystup>8 THEN
0014	vystup:=1;
0015	ok:=1;
0016	ELSE
0017	ok:=0;
0018	END_IF

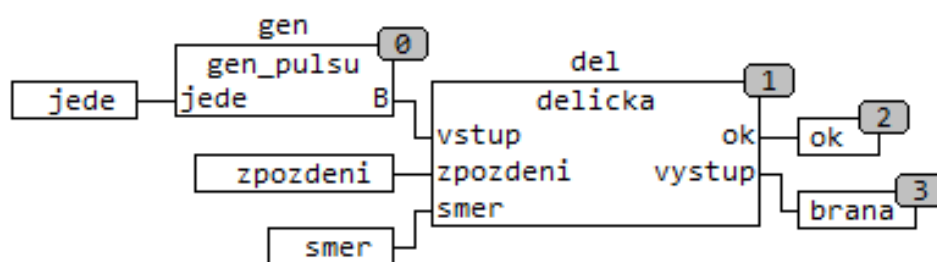
Obr. 1.5 - Strukturovaný text

Sekvenční funkční diagram SFC – Další grafický jazyk SFC (Sequential Function Chart) popisuje sekvenční chování programu. Skládá se z kroků a přechodů zobrazených v síti. Každý krok reprezentuje stav řízeného systému a je k němu přiřazen blok akcí. Přechod pak určuje podmínky, jež musí být splněny, aby mohl být aktivován krok, který přechodu předchází. [Martinásková, M., 2004]



Obr. 1.6 – Sekvenční funkční diagram [Vojáček, A., 2011]

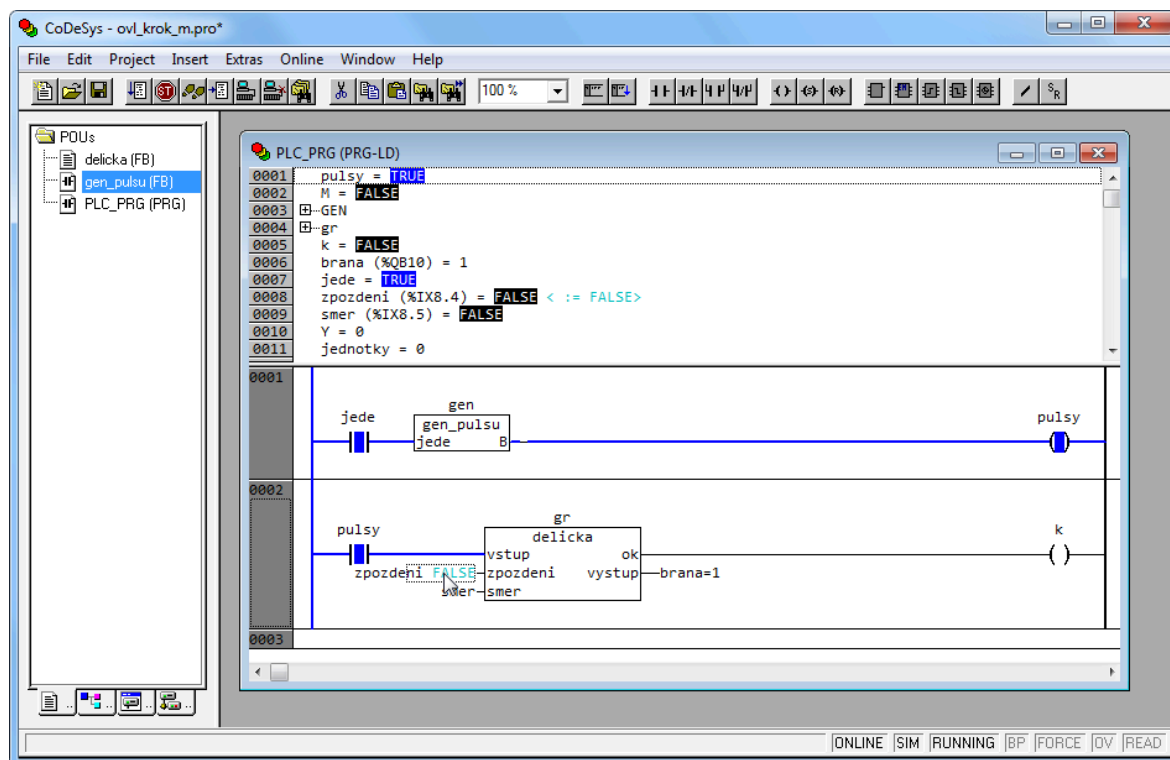
Kontinuální funkční diagram CFC – Grafický jazyk CFC (Continuous Function Chart) je velice podobný jazyku FBD. Na rozdíl od něj však není omezen liniovou strukturou, ale je zde volná plocha, na které lze jednotlivé bloky libovolně uspořádat a spojovat. [Martinásková, M., 2004]



Obr. 1.7 - Volně propojované bloky

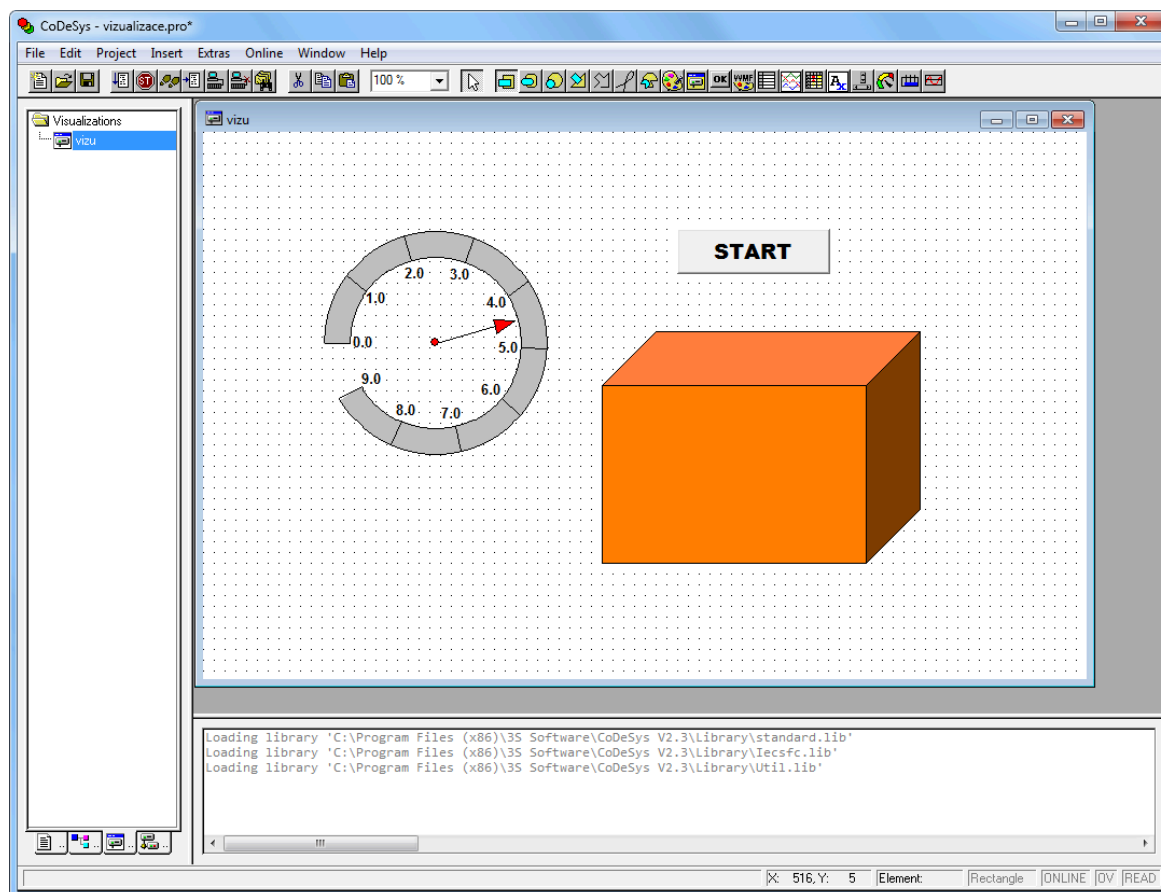
1.2 Monitorování

V programu CODESYS je možné probíhající procesy sledovat v reálném čase přímo ve vytvořených programech, kde se zobrazují hodnoty proměnných, viz Obr. 1.8. Vstupní proměnné je pak možné měnit i při spuštění programu.



Obr. 1.8 - Monitorování hodnot proměnných v programu CODESYS

Je rovněž možné vytvořit grafickou vizualizaci těchto procesů, viz obr. 1.9. K tomu slouží karta vizualizace na levé straně programovacího prostředí, kde se přidává vizualizační podprogram. Po vytvoření v něm pak lze vytvářet různé objekty. Ke všem objektům se potom dají přiřadit proměnné z vytvořených algoritmů, které mohou zajišťovat jejich případný pohyb, změnu jejich rozměrů a polohy, ovládání běžícího programu, změnu barvy nebo viditelnosti či prosté textové zobrazení hodnot proměnných z algoritmu. Objekty vybíráme z lišty nástrojů. Jsou zde v nabídce různé geometrické tvary objektů, tlačítka, tabulky, grafy atd.



Obr. 1.9 – Vizualizace v programu CODESYS

2 CNC výukové kity The Cool Tool

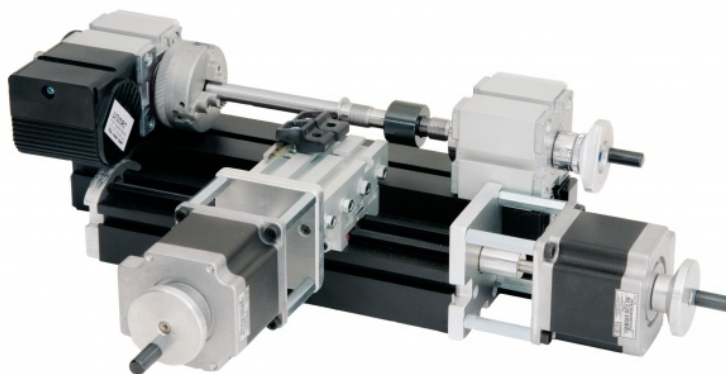
CNC výukové kity The Cool Tool jsou nástroje pro výuku automatického ovládání obráběcích strojů. Díky své modularitě lze vytvořit nepřeberné množství obráběcích strojů. V České Republice je nabízí např. firma HELAGO-CZ. V její nabídce jsou kromě CNC zařízení spolu s příslušenstvím i nástroje na jejich ovládání a ovládací software. Těmi se však zabývat nebudu, neboť je v projektu nepoužiju.

2.1 CNC modely dostupné na katedře

Zde je možné vidět CNC modely obráběcích strojů, které jsou dostupné na katedře automatizační techniky a řízení.

2.1.1 Unimat celokovový CNC soustruh

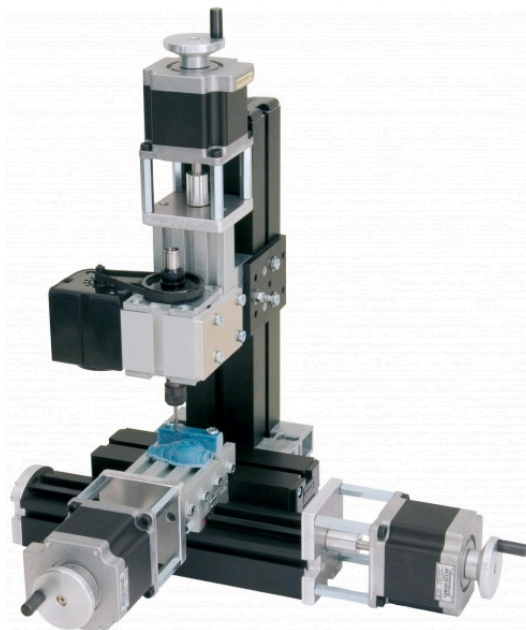
Základní vzdálenost mezi středy je 135 mm a točený průměr 49,5 mm, avšak obojí lze dokoupením komponent zvětšit. Soustruh obsahuje tříčelistové sklíčidlo s frézovanou spirálou, rotující hrot, celokovový vřeteník a koník, celokovové sáně, soustružnický nůž, držák a výškově stavitelné předlohy. Součástí jsou krokové motorky, buď 2, nebo 3 u soustruhu s doplňkovou osou. [HELAGO-CZ, 2012]



Obr. 2.1 - Unimat celokovový CNC soustruh [HELAGO-CZ, 2012]

2.1.2 Unimat celokovová 3-osá CNC frézka

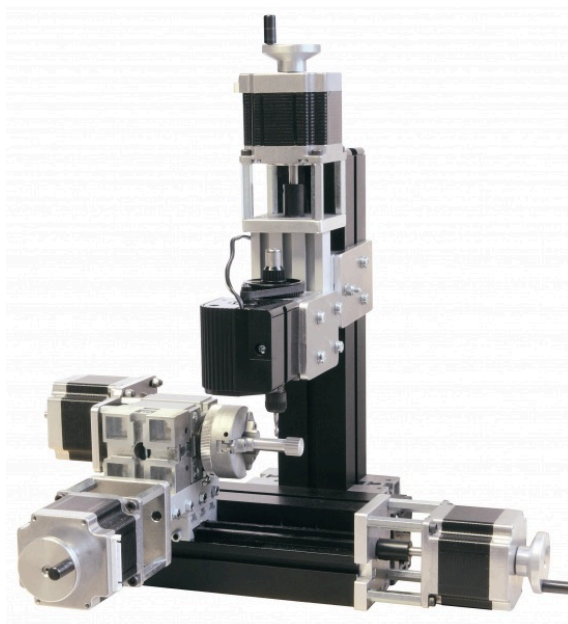
Její pracovní oblast je 145 × 50 × 50 mm. Obsahuje celokovové saně a vřeteník. Frézka obsahuje 3 krokové motorky, kabely a 14 V adaptér. Na katedře je dostupná vertikální frézka, nicméně přestavením modelu jí lze přestavět i na horizontální.



Obr. 2.2 - Unimat celokovová 3-osá CNC horizontální frézka [HELAGO-CZ, 2012]

2.1.3 Unimat celokovový 4-osý CNC frézovací systém

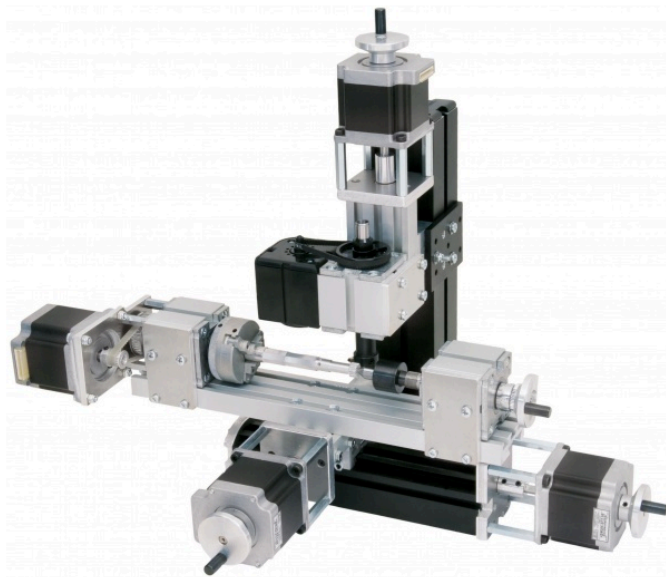
Jedná se o vertikální frézku s rotační osou. Tento systém na katedře sestavený není, ale opět by bylo možné, díky modularitě těchto kitů, jej z ostatních modelů sestavit. Její pracovní oblast je opět $145 \times 50 \times 50$ mm. Tento systém obsahuje vřeteník, 3 celokovové saně, rotační osu, 4 krokové motorky, kabely a 14 V adaptér.



Obr. 2.3 - Unimat celokovový 4-osý CNC frézovací systém [HELAGO-CZ, 2012]

2.1.4 Unimat celokovová 4-osá turn/mill sestava

Tato sestava je určena pro kombinaci soustružení a frézování na jednom nástroji. Obsahuje všechny části pro soustruh, soustruh s doplňkovou osou, vertikální i horizontální frézku a 4-osou frézku. Dále obsahuje 4 krokové motorky, kabely a 14 V adaptér.



Obr. 2.4 - Unimat celokovová 4-osá turn/mill sestava [HELAGO-CZ, 2012]

2.1.5 Portálová frézka Unistep 540/70

Tato fréza může sloužit k výrobě modelů, k frézování a rytí a k vrtání desek s plošnými spoji. Její pracovní oblast je $510 \times 308 \times 70$ mm. Posuv je realizován třemi krokovými motorky.



Obr. 2.5 - Portálová frézka Unistep 540/70 [HELAGO-CZ, 2012]

2.2 Pohonná jednotka - krokový motor

Krokový motor je druh elektrického pohonu, kde postupným řízeným napájením cívek dochází k magnetické indukci, jež otáčí rotorem. Jeho výhodou je možnost přesného úhlu natočení.

2.2.1 Rozdělení dle konstrukčního provedení

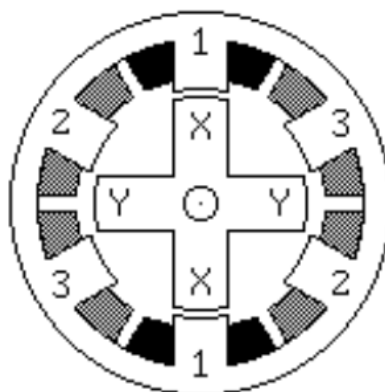
Reakční krokové motory – Jedná se o řešení, při kterém je rotor pasivní. Využívá proměnlivé magnetické vodivosti. Podmínkou funkčnosti takového motoru je rozdílný počet pólů na statoru a rotoru.

Krokové motory s aktivním rotorem – Rotor obsahuje magnetický aktivní část, tedy buď budící vinutí, nebo permanentní magnet. Permanentní magnet udržuje klidovou polohu, i když je statorové vinutí bez proudu.

Skupina slučující obě předchozí řešení

2.2.2 Princip krokového motoru

Na Obr. 2.6 je možné vidět reakční krokový motor. Stator obsahuje tři cívky a rotor obsahuje dvě dvojice zubů. Na obrázku je motor ve stavu, kdy proud protéká cívkou 1, která k sobě přitahuje dvojici zubů X. V případě, že se budící proud přepne na cívku 2, cívka k sobě přitáhne nejbližší dvojici zubů, tedy dvojici Y, a rotor se otočí o 30° . Stejným způsobem motor funguje, když se proud přepne na cívku 3, atd. [Tůma, J., Wagnerová, R., Farana, R. a Landryová, L., 2007]



Obr. 2.6 - Reakční krokový motor [Tůma, J., Wagnerová, R., Farana, R. a Landryová, L., 2007]

2.2.3 Funkční části CNC frézky

Jako úloha pro praktickou část této práce bylo vybráno ovládání 4-osé turn/mill sestavy. Ovšem vzhledem k tomu, že zadání práce se týká tříosého systému, není dále uvažováno ovládání čtvrté rotační osy. Z tohoto důvodu zde není o stroji hovořeno jako o turn/mill sestavě, ale jako o frézce.

Krokové motorky použité frézky

- napětí – až 24 V
- maximální odebíraný proud jednou cívkou – 2 A DC
- úhel kroku – $1,8^\circ$

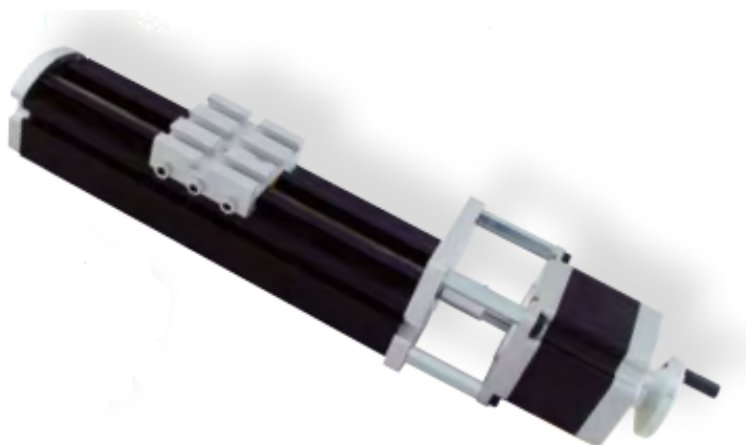
Šroubová tyč

- stoupání závitu – 1 mm

Motorek pohánějící frézu

- napětí – 24 V DC
- proud – až 3 A
- otáčky – 12000 rpm

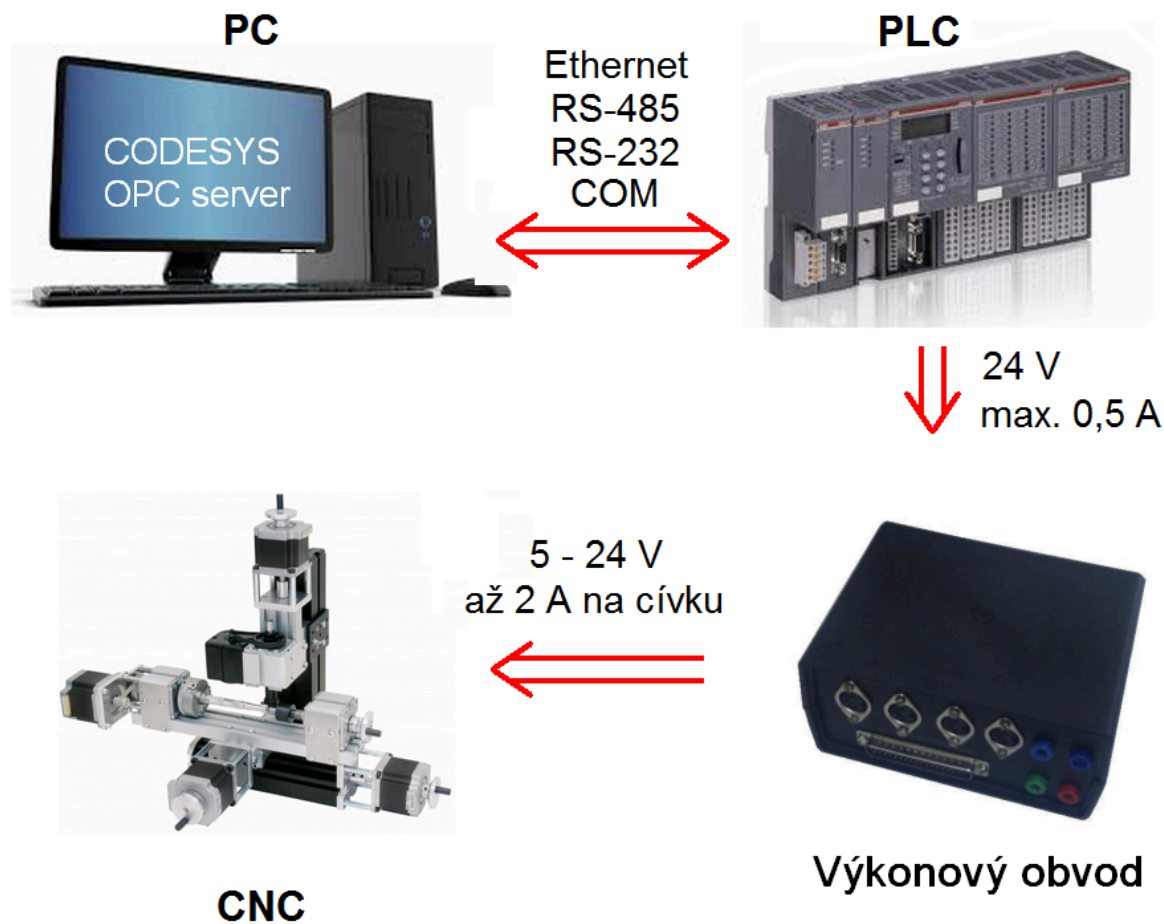
Krokové motorky této frézky jsou připojeny k šroubovým tyčím, jež následně pohybují s přemísťovanými objekty vsazenými do saní. Z výše uvedených údajů vyplývá, že za jednu otáčku krokového motorku, tedy 200 kroků, se přemísťovaný objekt posune o 1 mm. [The Cool Tool GmbH, 2012]



Obr. 2.7 - Polohovací mechanismus kitů The Cool Tool [The Cool Tool GmbH, 2012]

3 Propojení PLC, PC a frézky

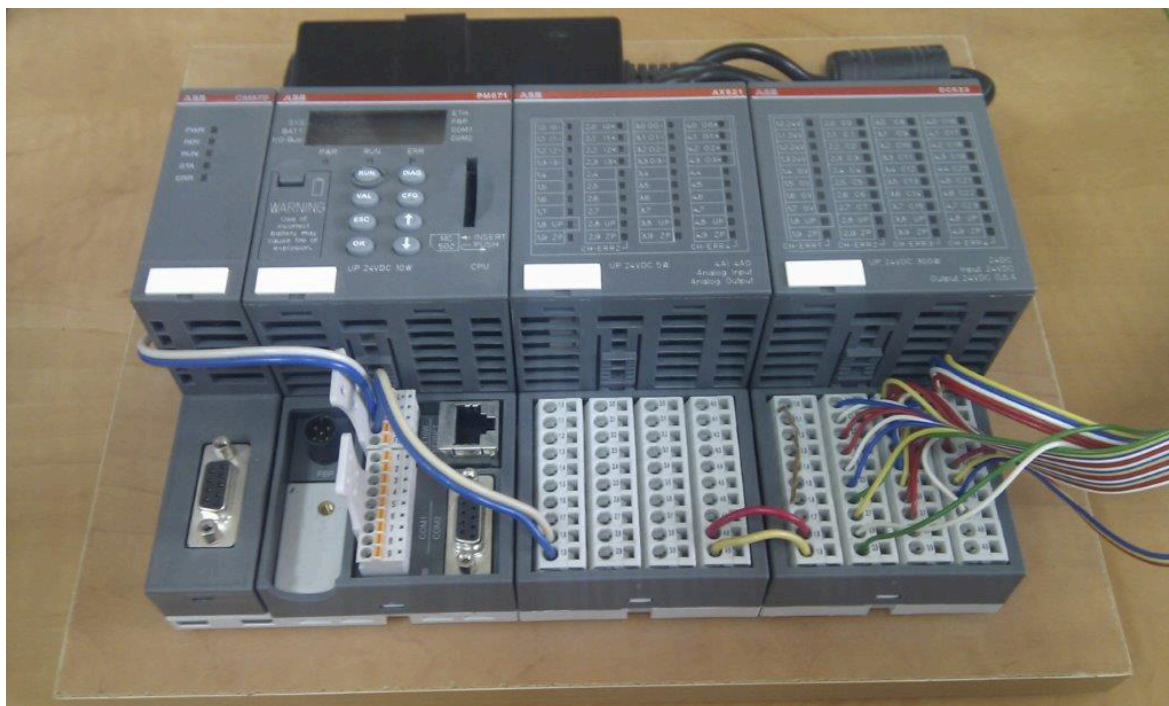
Celý systém se skládá z PC, PLC, spínacího výkonového obvodu a z ovládané frézky. Možnosti propojení je možné vidět na obr. 3.1.



Obr. 3.1 - Možnosti propojení

3.1 PLC řady AC500

V práci bylo použito PLC firmy ABB řady AC500. Řadí se mezi modulární PLC. V průmyslu je tato řada velice rozšířená díky svému výkonu, kvalitě a spolehlivosti. V nabídce firmy je velké množství rozšiřovacích modulů, takže v závislosti na konkrétních požadavcích lze sestavit různé kombinace PLC.



Obr. 3.2 – Použité PLC

Konkrétní typ použitého PLC je PM571. K PLC je mimo jiné připojen I/O modul DC523, který obsahuje 24 digitálních pinů, které můžou sloužit jako vstupy i jako výstupy. Právě z tohoto modulu byly použity výstupy pro ovládání krokových motorků. [Vaeprosys, 2010]

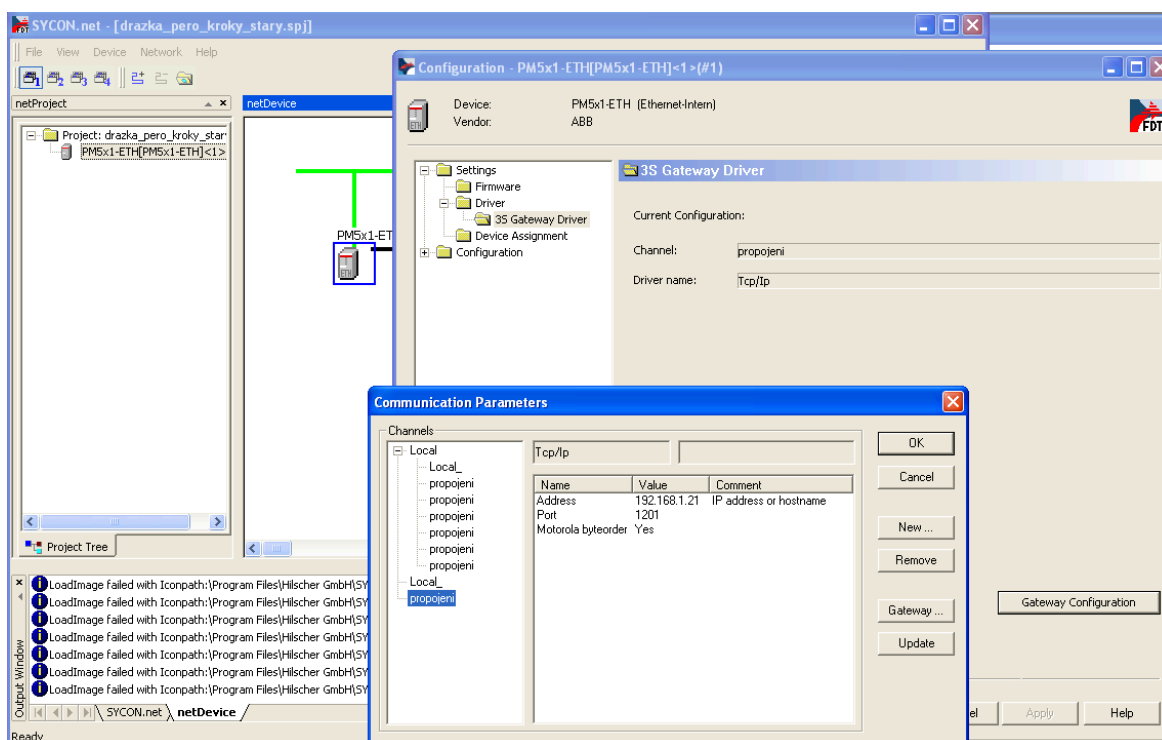


Obr. 3.3 - I/O modul pro PLC firmy ABB [Vaeprosys, 2010]

3.2 Propojení PLC s PC

Co se hardwarové cesty týče, může být komunikace PLC s počítačem řešena buď přes ethernet, RS-485, RS-232 nebo přes COM port. V této konkrétní úloze je PLC propojeno s PC přes ethernet. [Vaeprosys, 2010]

Propojení na softwarové úrovni se provádí v programu CODESYS. K PLC musí být přiřazena IP adresa, pomocí které pak s PC komunikuje. Po nakonfigurování použitého PLC a jeho modulů musí být v nástroji SYCON.net nastaveno konkrétní spojení programu s PLC. Je tedy třeba u ethernetového připojení napsat do komunikačních parametrů IP adresu onoho PLC a poté i přiřadit PLC v záložce Device Assignment.



Obr. 3.4 - Konfigurace v nástroji SYCON.net

S PLC řady AC 500 se dá komunikovat i prostřednictvím jiných softwarů, které nejsou defaultně určeny pro řízení PLC. Těmi je pak možno komunikovat s PLC pomocí OPC serveru. Ovšem vzhledem k tomu, že nebylo třeba využít žádného z takovýchto softwarů, nebylo třeba ani použít ke komunikaci OPC server.

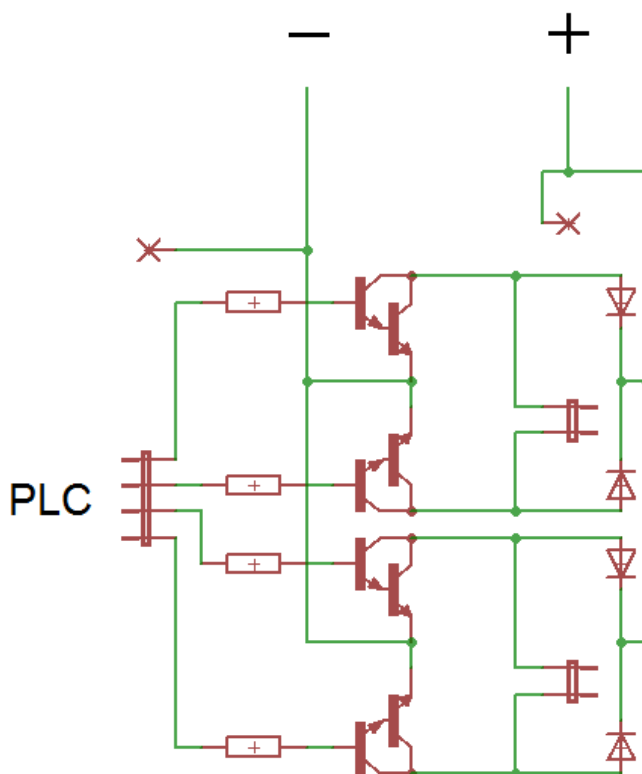
3.3 Propojení frézky s PLC

Jeden krokový motorek má čtyři cívky. Každá cívka pak představuje jeden bit. Z toho vyplývá, že na ovládání jednoho krokového motorku potřebujeme celkem čtyři digitální výstupy z PLC.

3.3.1 Návrh spínacího výkonového obvodu

I/O moduly firmy ABB mají maximální proudovou zátěž na jeden výstup 0,5 A. Takže pokud budeme chtít propojit krokové motorky s PLC, bude zapotřebí mít mezi nimi spínací výkonový obvod. K návrhu obvodu a plošného spoje byl použit program Eagle.

Princip tohoto obvodu je následující. Když je z PLC přiveden elektrický proud do tranzistoru, tranzistor se otevře a spojí se výkonový obvod obsahující příslušnou cívku krokového motorku. Cívka motorku představuje indukční zátěž, proto je třeba, aby paralelně s cívkou byla zapojena i ochranná dioda.

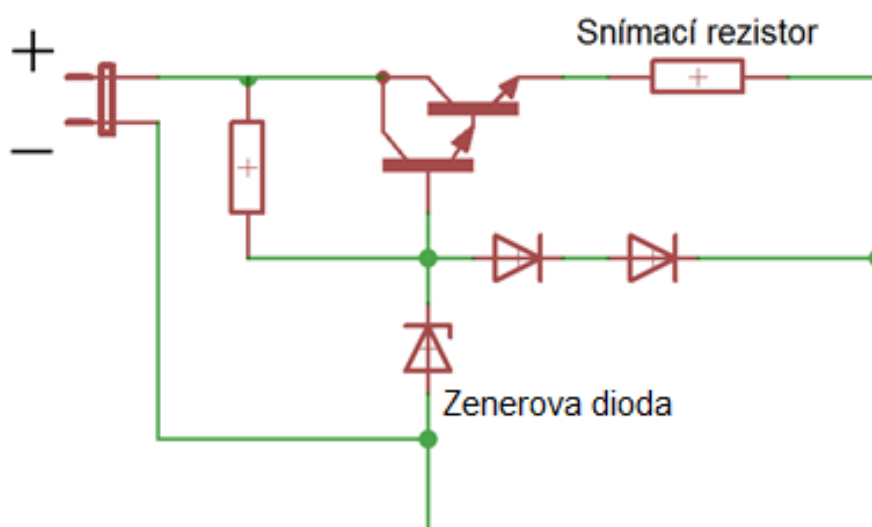


Obr. 3.5 - Spínací výkonový obvod

3.3.2 Proudové omezení krokového motorku

Vzhledem k tomu, že jedna cívka je schopná odběru 2 A, je zapotřebí vytvořit proudové omezení tak, aby tento proud nebyl překročen, viz obr. 3.6. Obvod využívá

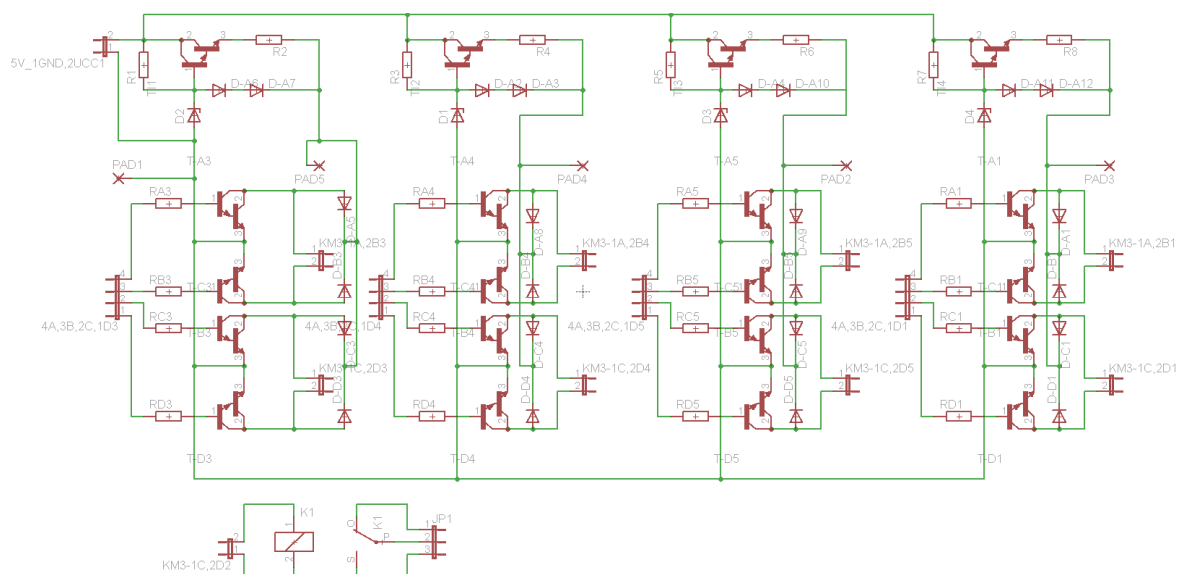
měření rozdílů úbytku napětí na snímacím rezistoru a dvou diodách. Při malém proudu dochází na rezistoru pouze k malému úbytku napětí, které funkci obvodu neovlivňuje. Se zvyšujícím se proudem roste i úbytek napětí na rezistoru. Pokud nastane situace, že je součet úbytku napětí na rezistoru a na polovodičovém přechodu báze-emitor větší než úbytek na diodách, začne diodami protékat proud, který omezuje napětí na tranzistoru. Tranzistor se tedy začne zavírat a omezí tím výstupní proud. Hodnota snímacího odporu v tomto případě vyšla $0,1 \Omega$. Zenerova dioda v obvodu pak stabilizuje napětí. V tomto konkrétním případě na přibližně $5,5 \text{ V}$. Nastavením velikosti napětí na zdroji pak lze měnit proud, který proudové omezení propustí. [Pandatron, 2012]



Obr. 3.6 - Proudové omezení

Pro napájení všech čtyř motorků musí celkový obvod obsahovat spínací výkonový obvod a proudové omezení čtyřikrát, pro každý motorek zvlášť. Součástí obvodu je i přepínací relé, které zapíná či vypíná motorek pohánějící frézu. Na ovládání krokových motorků je použito nižší napětí, než jakým je možné krokové motorky napájet, nicméně pro účely realizování laboratorních úloh na katedře je toto napětí dostatečné. Pro simultánní provoz všech čtyř motorků by bylo vhodné, aby použitý zdroj měl proudovou zatížitelnost až 8 A . V této úloze se však najednou točí maximálně dva motorky, takže k napájení stačí zdroj s poloviční zatížitelností.

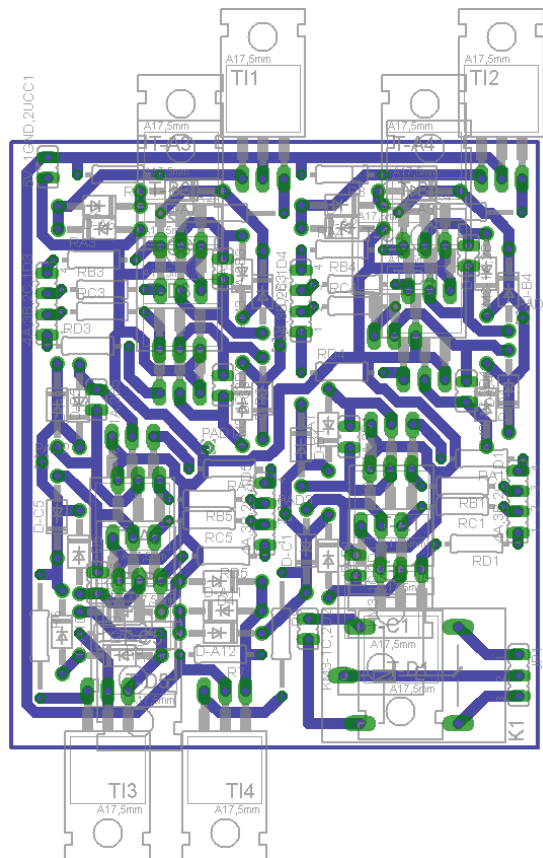
Seznam použitých součástek je v příloze A.



Obr. 3.7 – Celkové schéma spínacího výkonového obvodu

3.3.3 Tvorba plošného spoje

Po navržení obvodu bylo třeba v programu Eagle vytvořit plošný spoj. V tomto prostředí se zobrazí jednotlivé součástky a naznačené jejich spojení a úkolem je tyto součástky urovnat a spojit tak, aby se jednotlivé spoje nekřížily.



Obr. 3.8 – Rozmístění součástek na plošném spoji v programu Eagle

Co se fyzické realizace týče, je třeba nejprve na desku pokrytou mědí, která je navíc pokrytá fotocitlivou vrstvou, UV zářením ozářit spoj vytisknutý na průsvitný papír. Deska se pak ponoří do vývojky, ve které se odstraní ozářená fotocitlivá vrstva. Po opláchnutí se pak deska položí spojem na hladinu leptacího roztoku, který vyleptá měď, z níž byla odstraněna fotocitlivá vrstva. Výsledkem je, že měď zůstane na místech podle návrhu plošného spoje. Po odleptání mědi a po opláchnutí desky a očištění lihem se na desku nanese vrstva laku, který zabraňuje oxidaci a zároveň zjednodušuje následné pájení.

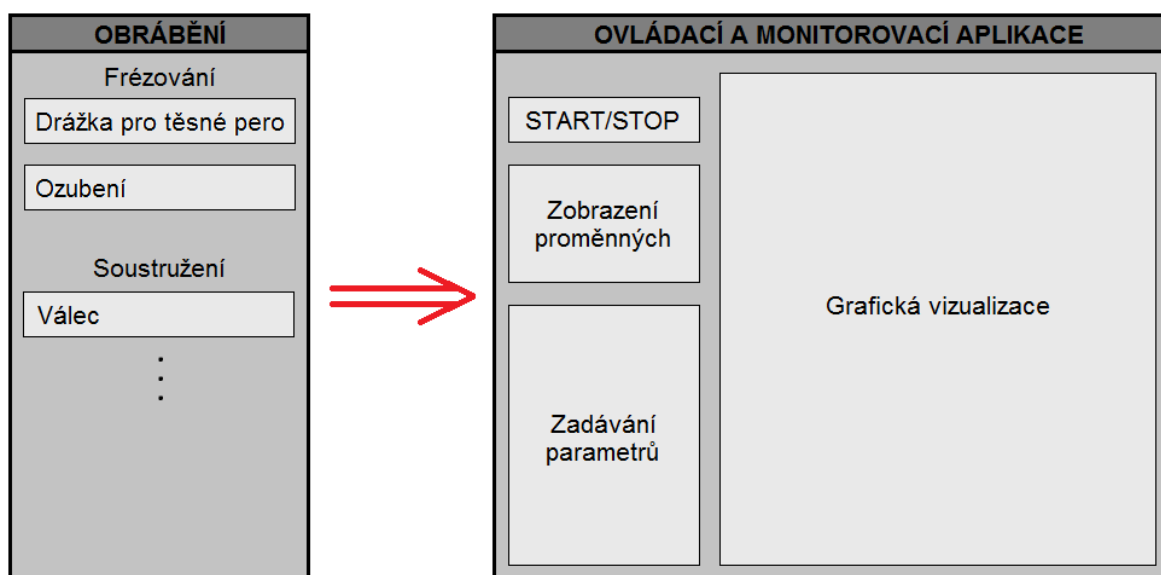
Poté co je plošný spoj vytvořený, je třeba do něj vyvrtat díry na součástky. Kvůli jednoduchosti následného pájení je vhodné, aby díry byly široké tak, aby jimi těsně prošla součástka. Nakonec, jakmile jsou díry vyvrtány, je třeba osadit spoj součástkami a ty připájet. [Kákona, J., Lafata, J., Horkel, M., 2013]

4 Řešení algoritmu pro vybranou úlohu

V algoritmu je nutné vyřešit, jakým způsobem budou přiváděny pulsy do krokových motorků a jakým způsobem budou řízeny tak, aby se motorky pohybovaly dle požadované trajektorie. I přes fakt, že se zadání týká tříosého mechanismu, je v programu, mimo hlavní algoritmus, vytvořeno ruční ovládání čtvrtého motorku.

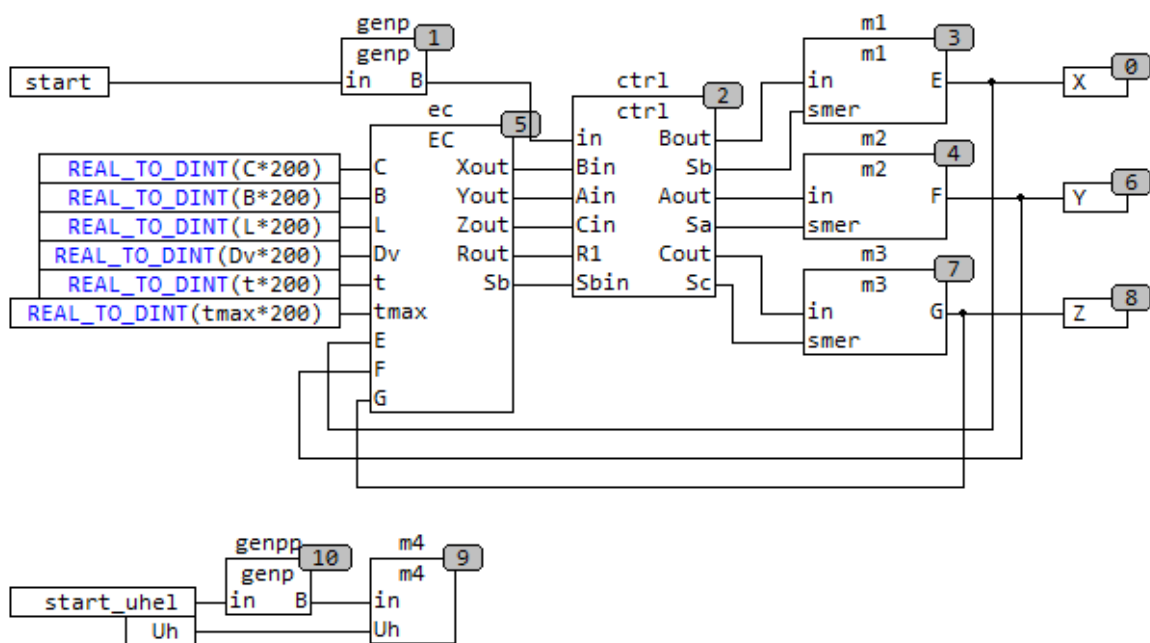
4.1 Systémový návrh

Představa systému je taková, že bude vytvořena aplikace pro více CNC obráběcích zařízení. Tato aplikace pak bude obsahovat seznam možností obrábění na jednotlivých strojích. Po kliknutí na určitou položku v seznamu se pak otevře příslušná část aplikace obsahující ovládací a monitorovací prvky.



Obr. 4.1 - Systémový návrh ovládací aplikace

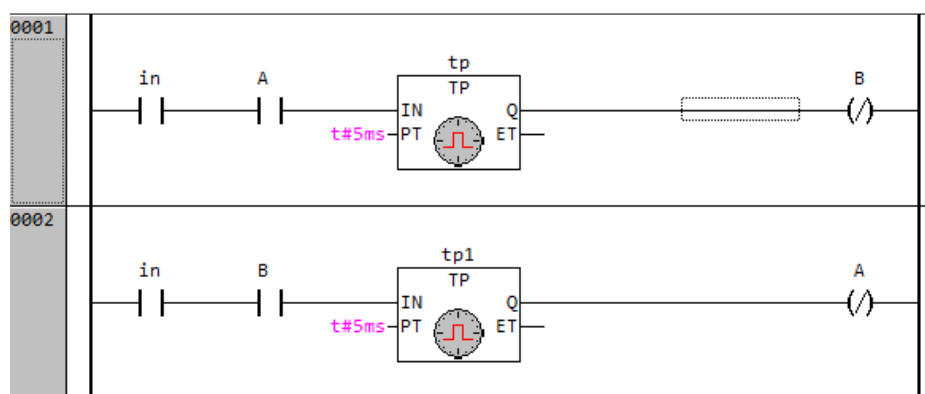
V této práci je vytvořena část aplikace pro frézování drážky pro těsné pero.



Obr. 4.2 - Blokové schéma řídicího algoritmu pro frézování drážky pro těsné pero

4.2 Generátor pulsů pro krokové motorky

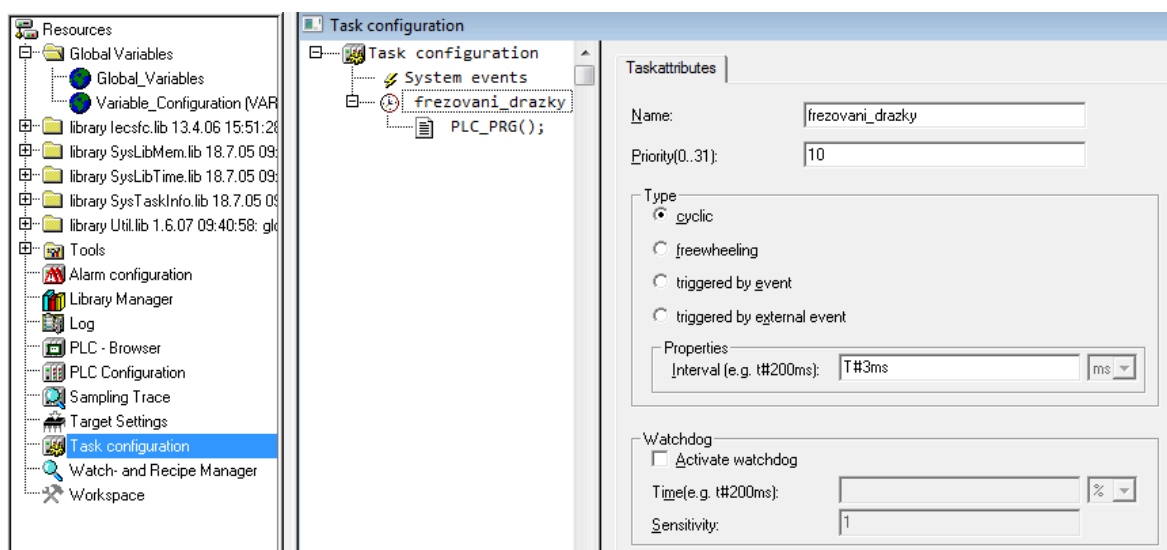
Pohyb krokového motorku je zajišťován vstupními pulsy spínajícími jednotlivé cívky. Proto je třeba vytvořit generátor pulsů v podobě astabilního klopného obvodu viz Obr. 4.3. K tomu byly použity TP časovače. TP časovač má při hodnotě vstupu TRUE hodnotu výstupu TRUE, která se po daném čase přepne na FALSE. Při hodnotě výstupu FALSE se pak změní vstupní hodnota před druhým časovačem na TRUE a tímto způsobem se jednotlivé proměnné neustále přepínají. Vyvedením jedné z těchto proměnných na výstup tohoto bloku se pak generují pulsy. Perioda pulsů je pak rovna součtu zadaných časových konstant. Pulsy je také možné v programu CODESYS vytvořit pomocí funkčního bloku generátoru periodických funkcí. V tomto algoritmu však byla zvolena předchozí možnost.



Obr. 4.3 - Generátor pulsů

Při daném napětí a proudu by teoreticky bylo možné použít časové konstanty rovné 2 ms tak, aby nedocházelo k prokluzu motorků. Avšak při testování algoritmu docházelo i k cyklům jejichž výpočet trval 2-3 ms. Z toho důvodu byla zvolena časová konstanta 3 ms.

Při základním nastavení v programu CODESYS trvá jeden cyklus, i přes nižší časové konstanty časovačů, odhadem 15 ms. Pro zvýšení rychlosti motorků je tedy nutné na kartě Resources v nabídce Task configuration nastavit úlohu a trvání jednoho cyklu viz obr. 4.4.



Obr. 4.4 - Nastavení Task configuration

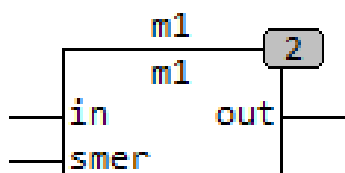
4.3 Pohyb krokového motorku

Při řešení pohybu krokového motorku je třeba uvážit dva aspekty. Zda se motorek otáčí a kterým směrem se otáčí. Každá cívka motorku představuje jeden bit, a tak je zapotřebí vytvořit smyčku čtyř bitů. To znamená, že při vstupním impulsu stačí vynásobit aktuální hodnotu výstupu dvěma, resp. vydělit dvěma při opačném směru otáčení. Po dosažení maximální, resp. minimální hodnoty, je pak třeba ještě přepnout hodnotu výstupu na počátek aktuálně prováděné smyčky.

Ještě je třeba ošetřit skutečnost, že během jednoho impulsu program provádí operaci několikrát. To se provede tak, že se do algoritmu zavede paměťová proměnná. Pak když je hodnota vstupu FALSE, hodnota paměti je TRUE. Po přepnutí vstupu na TRUE se provede operace, jen když je hodnota paměti TRUE. To znamená, že při provedení zbytku algoritmu se musí ještě hodnota paměti přepnout na FALSE. To zaručí, že po zbytek průběhu jednoho impulsu už program nebude danou operaci provádět.

Důležité je, aby při nulové rychlosti otáčení krokových motorků neprocházel cívkami žádný proud a zbytečně nezahříval součástky spínacího výkonového obvodu. To je vyřešeno pomocí TP časovačů tak, aby doba napájení jednotlivých cívek při jednom kroku byla rovna intervalu mezi dvěma pulsy.

Blok ovládající pohyb krokového motorku tedy musí obsahovat dva vstupy – vstupní signál přenášející pulsy a hodnotu směru otáčení.



Obr. 4.5 - Blok ovládající pohyb jednoho motorku

4.4 Řízení synchronního pohybu krokových motorků

Závislost počtu kroků synchronně se pohybujících motorků je dána funkčním předpisem. Uvažujme motorky M1 a M2. Vypočítává se hodnota funkce popisující počet kroků motorku M2 pro aktuální počet kroků motorku M1. Pokud pak počet kroků vykonaných motorkem M1 přesáhne odpovídající funkční hodnotu, pak motorek M2 vykoná krok. Problém však nastává, pokud se má motorek M1 pohybovat pomaleji než motorek M2. Bez patřičných úprav algoritmu by pak motorek M1 vykonával stejné množství kroků jako motorek M2. Aby k tomuto problému nedocházelo, je třeba přidat podmínku pro vykonání kroku motorku M1, že počet kroků motorku M2 musí být větší nebo roven než funkční hodnota popisující jeho pohyb pro aktuální počet kroků motorku M1. Když pak bude funkční hodnota menší, pohyb motorku M1 se v jednom cyklu algoritmu přeskočí, kdežto podmínka pro pohyb motorku M2 bude splněna. Při takto navrženém algoritmu je pak možno použít funkce, kde se střídají motorky v tom, který je rychlejší.

V algoritmu taky musí být řešeno, jakým směrem se má každý z motorků pohybovat. Zde se zjišťuje, jestli je zadaný počet kroků kladný nebo záporný, a podle toho se přiřadí k výstupní proměnné směru hodnota TRUE nebo FALSE. Poté už se v algoritmu počítá jen s absolutními hodnotami počtů kroků. Zde může být problém, když pohyb motorku zadaný funkčním předpisem mění směr (např. pohyb po kružnici). V takovém případě se musí funkce rozdělit na více částí, kde každá část je jen pro jeden směr pohybu.

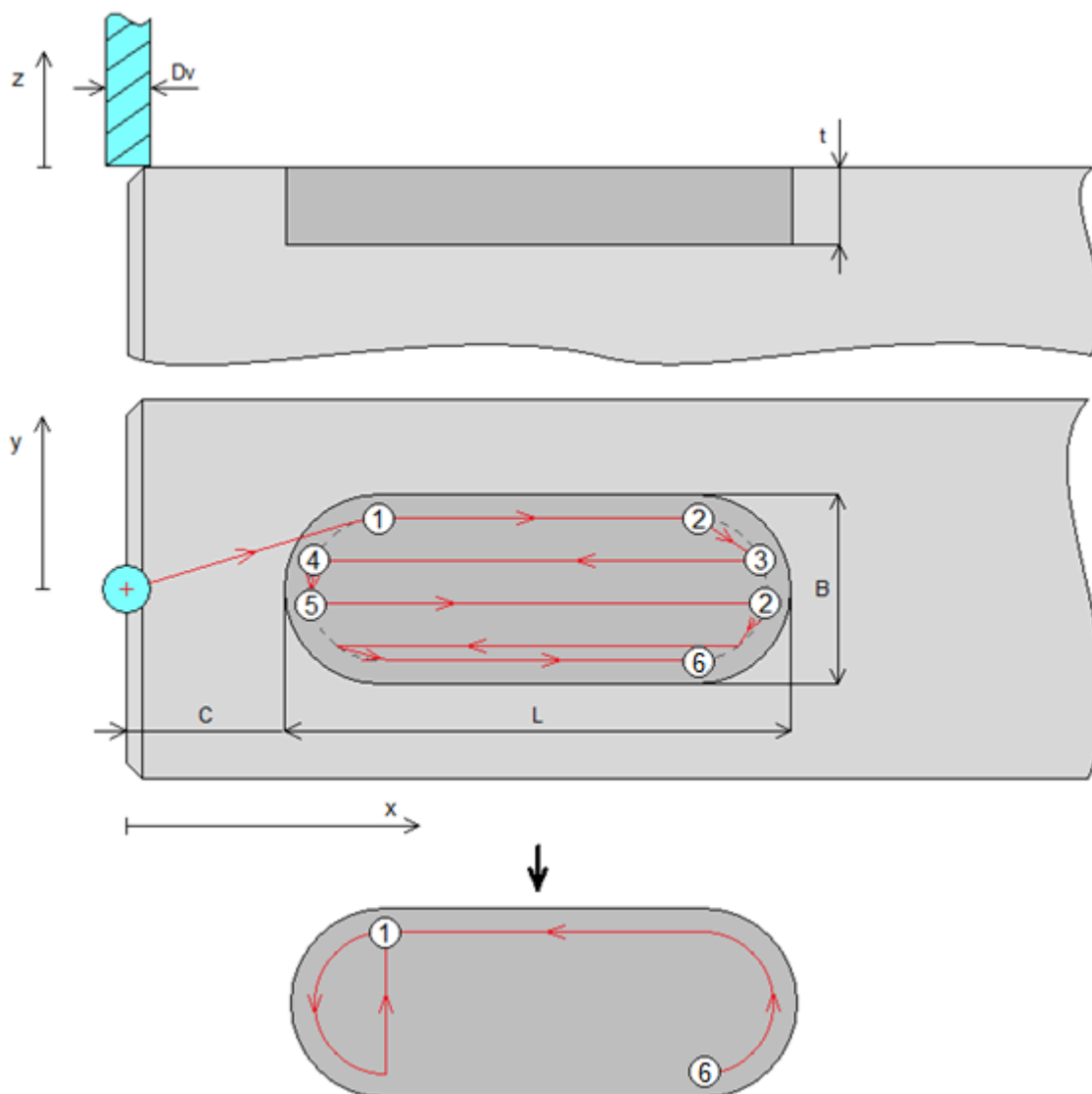
4.5 Algoritmus pro vyvrtání drážky pro těsné pero

V předchozích podkapitolách je popsán pouze jeden synchronní pohyb popsáný funkcí. Aby se však docílilo vyvrtání drážky pro těsné pero, je nutné zadat sadu takovýchto po sobě následujících pohybů. Proto se do celkového algoritmu musí přidat další blok, ve kterém je algoritmem popsána cesta obrobku. Z tohoto bloku se pak zadávají parametry do bloku řídicího synchronizovaný pohyb motorků.

Zadávané parametry:

- délka drážky – L [mm]
- šířka drážky – B [mm]
- vzdálenost drážky od kraje hřídele – C [mm]
- hloubka drážky – t [mm]
- maximální hloubka zářezu frézy – t_{max} [mm]
- průměr frézy – D_v [mm]

Počáteční poloha frézy vůči hřídeli je v ose x na kraji hřídele, v ose y na ose rotace hřídele a v ose z ve vzdálenosti poloměru hřídele od osy rotace. Tato poloha se před startem programu musí nastavit ručně. Ve frézce se pohybuje v ose x a v ose y obrobek a nástroj vykonává pouze vertikální a rotační pohyb, nicméně pro zjednodušení je popisována cesta frézy viz obr. 4.6.



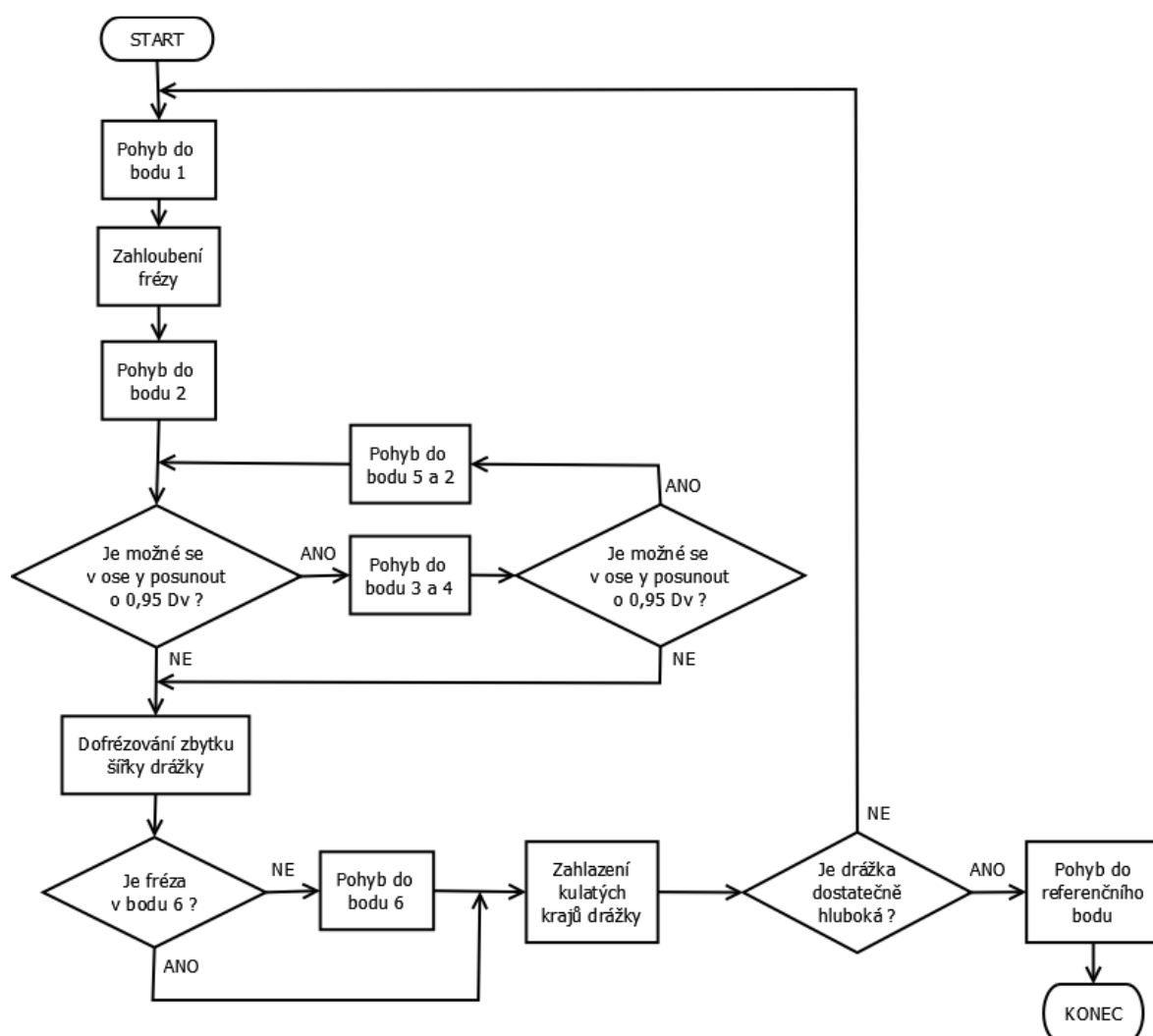
Obr. 4.6 - Pohyb frézy vůči obrobku

Před výpočet jednotlivých drah pohybu frézy jsou v algoritmu dány podmínky, popisující bod, ve kterém musí fréza být, aby došlo k určitému pohybu. V případě, že v průběhu vykonávání pohybů fréza projde přes určitý bod více krát, musí být přidány dodatečné podmínky jednoznačně určující další pohyb (např. paměťová proměnná).

Prvním krokem je, že se fréza přemístí do výšky 0,5 mm nad hřídel, aby při dalším přesunu nedošlo k poškození hřídele. Následně se přemístí do bodu 1. Tento bod musí být od hrany drážky ve vzdálenosti poloměru frézy. Jakmile se dostane do bodu 1, spustí se motorek ovládající frézu a nástroj se zahlubí na úroveň maximální hloubky zářezu. Následuje pohyb do bodů 2, 3, 4 a 5. Souřadnice bodů 2, 3, 4 a 5 jsou vypočítávány pomocí zadaných parametrů, přičemž v ose x se ještě připočítávají souřadnice půlkružnic. Rozdíly v souřadnicích na ose y při jednotlivých drahách pohybu po ose x jsou 0,95 krát průměr frézy. Díky tomu, že jsou body 2, 3, 4 a 5 vypočítávány obecně, lze vytvořit

smyčku, kde po bodu 5 bude následovat zase bod 2. V okamžiku, kdy už se fréza nemůže posunout v ose y o 0,95 násobek jejího průměru, posune se už jen o zlomek této hodnoty tak, aby byla dodržena zadaná šířka drážky.

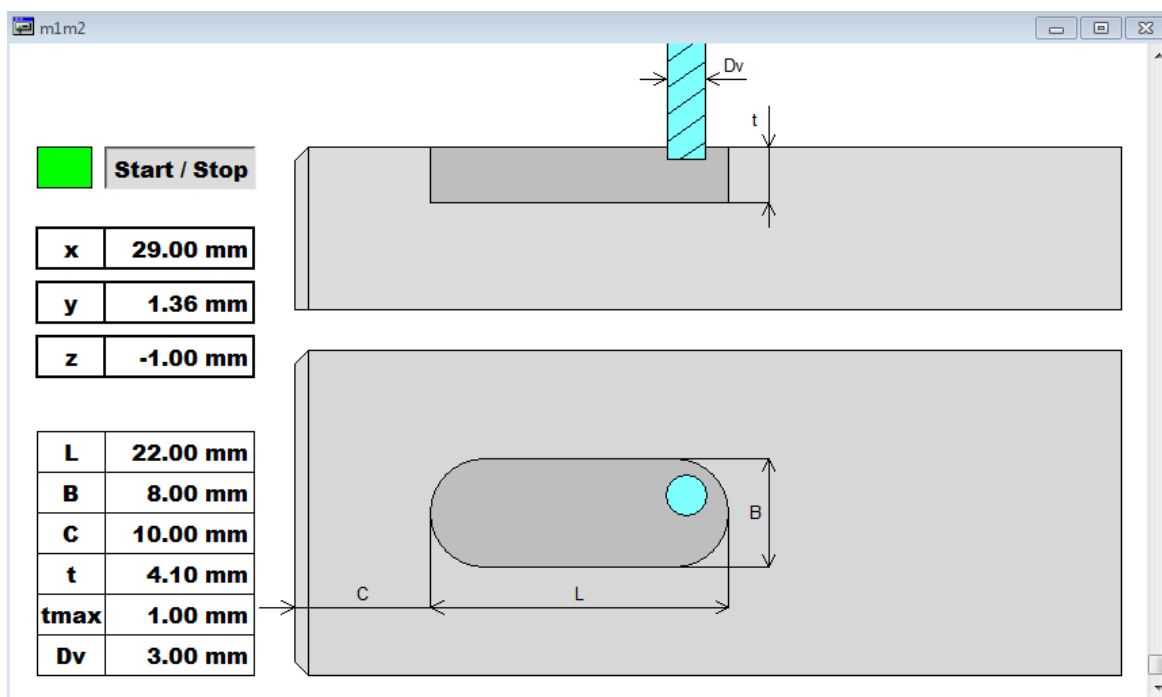
Jakmile se fréza dostane do bodu 6, přechází do fáze pohybu, při kterém kopíruje kulaté kraje drážky. Algoritmus je napsán tak, že k tomuto pohybu může dojít pouze z bodu 6. Pomocí paměťových proměnných je pak vyřešena situace, kdy fréza projde bodem 6, aniž by ještě vyfrézovala poslední část drážky. V takovém případě tuto část vyfrézuje a vrátí se do bodu 6. Po dokončení této fáze pohybu se porovnává, jestli už je drážka dostatečně hluboká. Pokud není, vrací se fréza zpět do bodu 1, kde začne frézovat další vrstvu drážky. Pokud už je drážka dostatečně hluboká, fréza se vrátí do počátečních souřadnic, přičemž se automaticky zastaví motorek zajišťující její pohon.



Obr. 4.7 - Vývojový diagram algoritmu pro pohyb frézy

5 Ovládací a monitorovací aplikace

Při frézování vzniká kolem drážky odpad, přes který nemusí jít vidět na proces frézování a na polohu nástroje, případně uživatel ani nemusí být fyzicky poblíž zařízení. Z tohoto důvodu a za účelem jednoduchého ovládání programu je vytvořena jednoduchá grafická ovládací a monitorovací aplikace.



Obr. 5.1 - Ovládací a monitorovací aplikace

Aplikace obsahuje grafické zobrazení hřídele, drážky pro těsné pero a frézy. Pro přehlednost jsou zde zobrazeny i všechny důležité rozměry. Pomocí posunutí a zvětšování či zmenšování objektů v závislosti na hodnotách parametrů a proměnných je realizován posuv frézy i grafické zobrazení drážky a frézy včetně rozměrů. Součástí aplikace jsou i číselně vyjádřené vzdálenosti v osách x, y a z. Před spuštěním programu je pak možnost zadat hodnoty parametrů drážky, které automaticky změní hodnoty proměnných v programu. Pro spuštění a zastavení programu pak slouží tlačítko Start/Stop.

6 Závěr

Cílem mé práce bylo řízení víceosého mechanismu pomocí PLC firmy ABB. Nejprve jsem se tedy musel seznámit s programovacím prostředím CODESYS, jež firma ABB ke svým PLC poskytuje. Poté jsem se seznámil s konkrétními CNC modely obráběcích strojů, které jsou dostupné na katedře automatizační techniky a řízení, a s možnostmi jejich propojení s PLC. Pro praktickou část této práce jsem vybral ovládání čtyřosé turn/mill sestavy. Konkrétně jsem tvořil algoritmus pro frézování drážky pro těsné pero.

PLC jsem propojil s PC, kde jsem pak v programu CODESYS nakonfiguroval připojení, tak aby správně probíhala komunikace mezi PC a PLC. I/O modul DC523, který jsem použil na propojení PLC s frézkou, má nedostatečnou proudovou zatížitelnost pro rozpohybování krokových motorků ovládaného mechanismu. Z tohoto důvodu jsem musel navrhnout a vyrobit spínací výkonový obvod. Na ovládání krokových motorků jsem použil nižší napětí, než jakým je možné krokové motorky napájet, nicméně pro účely realizování laboratorních úloh na katedře je toto napětí dostatečné.

Vytvořil jsem systémový návrh, jehož podstatou je vytvoření komplexní aplikace umožňující různé operace na různých CNC zařízeních. Tato aplikace bude obsahovat seznam různých možností obrábění, odkazující na příslušné ovládací a monitorovací části aplikace. V této práci jsem pak řešil část aplikace pro frézování drážky pro těsné pero.

Při návrhu algoritmu vyfrézování drážky pro těsné pero jsem tedy nejprve musel vymyslet část algoritmu, která realizuje samotný pohyb motorků. Zde jsem mimo jiné vyřešil situaci, kdy se jeden motorek zastavil, ale pořád odebíral proud. To bylo důležité, aby se nezahřívaly součástky. Poté jsem pomocí funkčních předpisů napsal algoritmus pro zadávání konkrétních synchronizovaných pohybů. Konečnou fází algoritmu pak bylo vyřešení zadávání jednotlivých pohybů tak, aby se na frézce podle zadaných parametrů vyfrézovala drážka.

Nakonec jsem k vytvořenému algoritmu vytvořil i ovládací a monitorovací aplikaci. Lze v ní v reálném čase přehledně sledovat, jak se fréza vůči obrobku v daném okamžiku pohybuje. Lze v ní zadávat jednotlivé parametry drážky, přičemž jejich změnou se také změní i rozměry a umístění drážky v grafické podobě. Do této aplikace jsem rovněž přidal ovládací tlačítka umožňující spuštění či zastavení programu.

Tímto řešením jsem ošetřil pouze jednu část možného obrábění pomocí vybrané frézky. Dalším řešením by tedy bylo vytvořit algoritmy pro jiné typy obrábění a tyto algoritmy pak vhodně vsadit do komplexnější aplikace.

Seznam použité literatury

- BOYER, S. A. 1999. *SCADA: Supervisory Control and Data Acquisition*, 2nd Edition. New York (USA): ISA, 1999. 215 p. ISBN 1-55617-660-0.
- CNC výukové kity [online]. HELAGO-CZ, s.r.o. © 2002-2012. [cit. 11.10.2012]. Dostupné z: <<http://www.helago-cz.cz/catalog/cnc-vyukove-kity/>>
- KÁKONA, J., LAFATA, J., HORKEL, M. Domácí výroba plošných spojů fotocestou. [online]. MLAB © 2013 [cit. 16.5.2013]. Dostupné z: <http://www.mlab.cz/Articles/HowTo/How_to_make_PCB/DOC/HTML/How_to_make_PCB.cs.html>
- MARTINÁSKOVÁ, M. Programovací jazyky pro PLC. *Automatizace* [online časopis]. 2004. roč. 47, č. 6 [cit. 10.12.2012]. ISSN 0005-125X. Dostupné z: <<http://www.automatizace.cz/article.php?a=142>>.
- PANDATRON. Externí proudový omezovač. *Pandatron.cz – Elektrotechnický magazín* [online]. 11.5.2012 [cit. 15.5.2013]. ISSN 1803-6007 Dostupné z: <http://pandatron.cz/?3247&externi_proudovy_omezovac>
- ŠOFER, P. *Vizualizační software pro monitorování a konfiguraci systémů s PLC*. Ostrava: katedra ATR – 352 VŠB-TUO, 2011, 66s. Diplomová práce, vedoucí: Tůma, J., konsultant: Škuta, J.
- ŠTULPA, M. *CNC obráběcí stroje a jejich programování*. 1. vyd. Praha, BEN – technická literatura, 2006. 128 s. ISBN 80-7300-207-8.
- TŮMA, J., WAGNEROVÁ, R., FARANA, R. a LANDRYOVÁ, L. *Základy automatizace*. 1. vyd. Ostrava, Ediční středisko VŠB – TUO, 2007. [cit. 7.2.2013]. 280s. ISBN 978-80-248-1523-7
- VAEPROSYS - AUTOMATIZACE - PLC AC500 - Dokumentace ke stažení [online]. VAE ProSys s.r.o. © 2010. [cit. 11.10.2012]. Dostupné z: <<http://vaeprosys.cz/list.asp?p=133&k=32&mk=9&ms=32>>
- VLACH, J. *Počítačová rozhraní, přenos dat a řídicí systémy*. Praha, BEN-technická literatura, 1997, ISBN 80-85940-17-4.

VOJÁČEK, A. Programovací režimy pro PLC dle IEC 61131-3 (CoDeSys). *Automatizace.HW.cz* [online]. HW server s.r.o., 2011 [cit. 10.12.2012]. Dostupné z: <<http://automatizace.hw.cz/programovaci-rezimy-pro-plc-dle-iec-611313-codesys>>

WHITT, M. D. 2003. *Successful Instrumentation and Control Systems Design*. New York (USA): ISA, 2003. 360 p. ISBN 1-55617-844-1.

TheCoolTool Product Catalogue [online]. The Cool Tool GmbH © 2012. [cit. 7.2.2013]. Dostupné z: <http://www.unimateducation.com/fileadmin/images/uids/downloads/VS-CAT_06-11-2012_WEB.pdf>

Přílohy

Příloha A – Seznam součástek ve spínacím výkonovém obvodu

Příloha B - Pohyb krokového motorku

Příloha C – Část algoritmu pro synchronní pohyb krokových motorků

Příloha D – Část algoritmu pro vyvrtání drážky pro těsné pero

Příloha A – Seznam součástek ve spínacím výkonovém obvodu

Rezistory	
R1, R3, R5, R7	100 Ω
R2, R4, R6, R8	0,1 Ω (2 W)
RA1, RA3, RA4, RA5, RB1, RB3, RB4, RB5, RC1, RC3, RC4, RC5, RD1, RD3, RD4, RD5	3K7 Ω
Diody	
D1, D2, D3, D4	1N5339
D-A1, D-A2, D-A3, D-A4, D-A5, D-A6, D-A7, D-A8, D-A9, D-A10, D-A11, D-A12	1N4007
Tranzistory	
TI1, TI2, TI3, TI4	BDX33C
T-A1, T-A3, T-A4, T-A5, T-B1, T-B3, T-B4, T-B5, T-C1, T-C3, T-C4, T-C5, T-D1, T-D3, T-D4, T-D5	TIP121

Příloha A - Pohyb krokového motorku

```
IF in THEN (*vstupní puls*)
  IF pam THEN (*paměť zaručující pouze jedno opakování cyklu v rámci jednoho
vstupního impulsu*)
    pam:=FALSE;
    IF smer=TRUE THEN (*rozhodnování o směru otáčení*)
      out:=out*2; (*výpočet dalšího bitu ve smyčce*)
      E:=E+1; (*celkový počet kroků v ose x*)
      IF out>8 THEN (*vrácení se na začátek smyčky*)
        out:=1;
      END_IF
    ELSE
      out:=out/2;
      E:=E-1;
      IF out<1 THEN
        out:=8;
      END_IF
    END_IF
    Ev:=E/20; (*výpočet proměnné pro vizualizaci*)
    CASE out OF (*přiřazení proměnných pro jednotlivé výstupní pulsy*)
      1: out11:=TRUE; out22:=FALSE; out33:=FALSE; out44:=FALSE;
      2: out22:=TRUE; out11:=FALSE; out33:=FALSE; out44:=FALSE;
      4: out33:=TRUE; out22:=FALSE; out11:=FALSE; out44:=FALSE;
      8: out44:=TRUE; out22:=FALSE; out33:=FALSE; out11:=FALSE;
    END_CASE
  END_IF
ELSE
  pam:=TRUE;
END_IF
(*časovače omezující dobu trvání výstupních pulsů*)
tpinst(in:=out11, PT:=t#10ms); out1:=tpinst.Q;
tpinst1(in:=out22, PT:=t#10ms); out2:=tpinst1.Q;
tpinst2(in:=out33, PT:=t#10ms); out3:=tpinst2.Q;
tpinst3(in:=out44, PT:=t#10ms); out4:=tpinst3.Q;
IF smer=FALSE THEN (*zamezení dvou pulsů najednou*)
  IF out2=TRUE THEN out1:=FALSE; END_IF
  IF out3=TRUE THEN out2:=FALSE; END_IF
  IF out4=TRUE THEN out3:=FALSE; END_IF
  IF out1=TRUE THEN out4:=FALSE; END_IF
ELSE
  IF out2=TRUE THEN out3:=FALSE; END_IF
  IF out3=TRUE THEN out4:=FALSE; END_IF
  IF out4=TRUE THEN out1:=FALSE; END_IF
  IF out1=TRUE THEN out2:=FALSE; END_IF
END_IF
```

Příloha B – Část algoritmu pro synchronní pohyb krokových motorků

```
(*vykonávání kroků motorku v ose y*)
IF A>0 AND co<A AND Da<=cna AND D<=cn AND D1<=cn1 AND D2<=cn2 THEN
    co:=co+1; (*čítač kroků*)
    Aout:=TRUE; (*vykonání impulsu pro posun motorku*)
END_IF

(*vykonávání kroků motorku v ose x*)
IF cn<B AND R1=0 THEN
    IF Bin>0 THEN (*určení směru otáčení motorku*)
        Sb:=TRUE;
    ELSE
        Sb:=FALSE;
    END_IF
    IF A>0 THEN (*rozhodování jestli je pohyb motorku na ose x závislý na pohybu v ose y*)
        D:=co*B/A; (*výpočet aktuálního počtu kroků na ose x v závislosti na počtu kroků na ose y*)
        IF D>cn THEN (*rozhodování, zda v daném cyklu dojde ke kroku*)
            Bout:=TRUE; (*vykonání impulsu pro posun motorku*)
            cn:=cn+1; (*čítač kroků*)
        END_IF
    ELSE
        Bout:=TRUE; (*vykonání impulsu pro posun motorku*)
        cn:=cn+1; (*čítač kroků*)
    END_IF
END_IF

(*vykonávání kroků motorku v ose z*)
IF cna<C THEN
    IF A>0 THEN (*rozhodování, jestli je pohyb motorku na ose z závislý na pohybu v ose y*)
        Da:=co*C/A; (*výpočet aktuálního počtu kroků na ose z v závislosti na počtu kroků na ose y*)
        IF Da>cna THEN
            Cout:=TRUE; (*vykonání impulsu pro posun motorku v ose z*)
            cna:=cna+1; (*čítač kroků*)
        END_IF
    ELSE
        Cout:=TRUE; (*vykonání impulsu pro posun motorku v ose z*)
        cna:=cna+1; (*čítač kroků*)
    END_IF
END_IF

(*pohyb po kružnici*)
IF co<R1 THEN (*výpočet pro první půlku půlkružnice*)
    Sb:=Sbin; (*určení aktuálního směru pohybu v ose x*)
    D1:=SQRT(EXPT(R1,2)-EXPT((co-R1),2)); (*výpočet aktuálního počtu kroků na ose x v závislosti na počtu kroků na ose y*)
    IF D1>cn1 THEN (*zjišťování, jestli má motorek v ose x vykonat další krok*)
        Bout:=TRUE; (*vykonání impulsu pro posun motorku v ose x*)
    END_IF
END_IF
```

```

        cn1:=cn1+1; (*čítač kroků*)
    END_IF
END_IF
IF 2*R1>=co AND co>=R1 AND cn1>0 THEN (*výpočet pro druhou půlku
půlkružnice*)
    IF Sbin=TRUE THEN (*opačný směr pro druhou půlku půlkružnice*)
        Sb:=FALSE;
    ELSE
        Sb:=TRUE;
    END_IF
    D2:=R1-SQRT(EXPT(R1,2)-EXPT((co-R1),2)); (*výpočet aktuálního počtu kroků na
ose x v závislosti na počtu kroků na ose y*)
    IF D2>cn2 THEN (*zjišťování, jestli má pomalejší motorek vykonat další krok*)
        Bout:=TRUE; (*vykonání impulsu pro posun motorku*)
        cn2:=cn2+1; (*čítač kroků*)
    END_IF
END_IF

```

Příloha C – Část algoritmu pro vyvrtání drážky pro těsné pero

```
(*posun do bodu 2*)
IF G=Z AND F=Y AND E=X THEN
    Rout:=0;
    X3:=-1;
    X1:=REAL_TO_DINT(C+L-B/2+SQRT(EXPT(R1,2)-EXPT(Y,2))); (*výpočet
souřadnice v ose x zohledňující funkční hodnotu bodu na kružnici v ose x pro aktuální
souřadnici na ose y*)
    Xout:=X1-X; Yout:=0; Zout:=0;
END_IF
(*posun do bodu 3*)
IF E=X1 THEN
    IF -R1<Y-0.95*Dv THEN (*posun, pokud je ještě možné posnout frézu o 0,95
násobek jejího průměru v ose y*)
        Y2:=REAL_TO_DINT(Y-0.95*Dv); (*posun v ose y o 0,95 násobek průměru
frézy*)
        X2:=REAL_TO_DINT(C+L-B/2+SQRT(EXPT(R1,2)-EXPT(Y2,2)));
        Xout:=X2-X1; Yout:=Y2-Y;
    ELSE (*posun, pokud již není možné posnout frézu o 0,95 násobek jejího průměru v
ose y*)
        Y2:=-R1; X2:=C+L-B/2;
        Xout:=X2-X1; Yout:=Y2-Y; K:=TRUE;
    END_IF
END_IF
(*posun do bodu 4*)
IF E=X2 AND F=Y2 THEN
    Rout:=0;
    X1:=-1;
    X3:=REAL_TO_INT(C+B/2-SQRT(EXPT(R1,2)-EXPT(Y2,2)));
    Xout:=X3-X2; Yout:=0;
END_IF
(*posun do bodu 5*)
IF E=X3 THEN
    IF -R1<Y2-0.95*Dv THEN
        Y:=REAL_TO_DINT(Y2-0.95*Dv); X:=REAL_TO_INT(C+B/2-
SQRT(EXPT(R1,2)-EXPT(Y,2)));
        Xout:=X-X3; Yout:=Y-Y2;
    ELSE
        Y:=-R1; X:=C+B/2;
        Xout:=X-X3; Yout:=Y-Y2; K1:=TRUE;
    END_IF
END_IF
```